4-2010

Generating synonyms based on query log data

Stelios PAPARIZOS

Tao CHENG

Hady W. LAUW
Singapore Management University, hadywlauw@smu.edu.sg

US 20100082657A1

(19) United States

(12) Patent Application Publication

(10) Pub. No.: US 2010/0082657 A1

(43) Pub. Date: Apr. 1, 2010

Paparizos et al.

(54) GENERATING SYNONYMS BASED ON QUERY LOG DATA

(75) Inventors: **Stelios Paparizos**, San Jose, CA (US); **Tao Cheng**, Urbana, IL (US); **Hady W. Lauw**, Mountain View, CA (US)

Correspondence Address:
MICROSOFT CORPORATION
ONE MICROSOFT WAY
REDMOND, WA 98052 (US)

(73) Assignee: **Microsoft Corporation**, Redmond, WA (US)

(21) Appl. No.: **12/235,635**

(22) Filed: **Sep. 23, 2008**

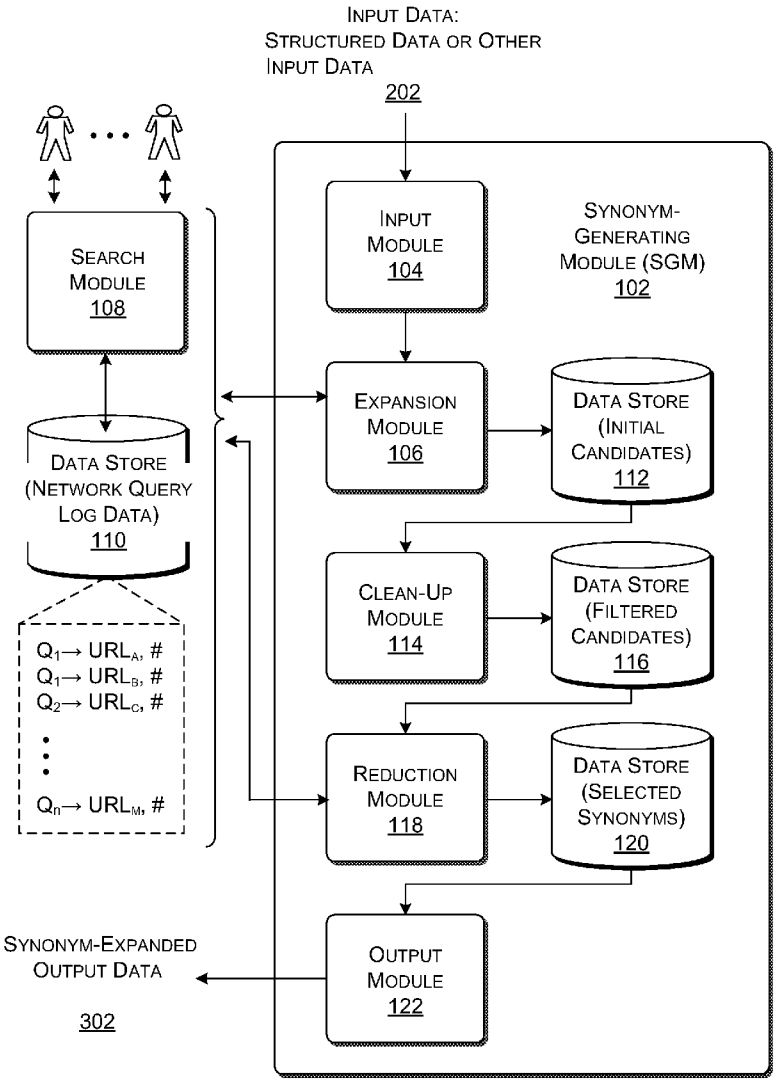
Publication Classification

(51) **Int. Cl.**
G06F 17/30 (2006.01)

(52) **U.S. Cl.** **707/767**; 707/E17.015

(57) **ABSTRACT**

An approach is described for generating synonyms to supplement at least one information item, such as, in one case, a set of related items. The approach can involve an expansion phase, a clean-up phase, and a reduction phase. In the expansion phase, the approach identifies, for each related item, a set of initial synonym candidates. In the clean-up phase, the approach removes noise from the set of initial synonym candidates (if such noise exists), to provide a set of filtered synonym candidate items. In the reduction phase, the approach ranks and applies a threshold (or thresholds) to the set of filtered synonym candidate items, to generate, for each information item, a set of selected synonyms. The approach uses query log data as at various points in its operation. The selected synonyms can be used to improve the effectiveness of user searches.



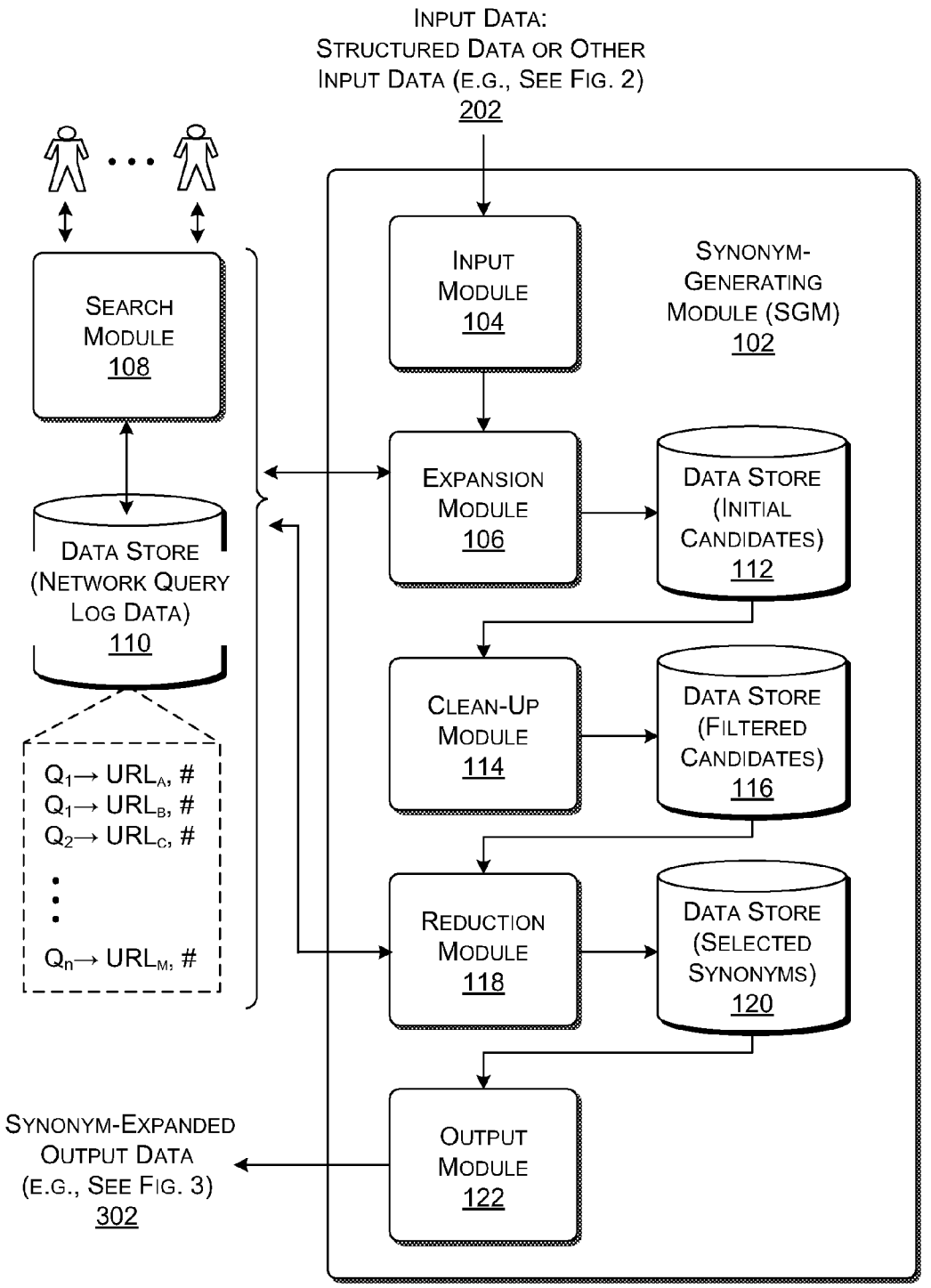

INPUT DATA:
STRUCTURED DATA OR OTHER
INPUT DATA (E.G., SEE FIG. 2)
202
SEARCH MODULE 108
DATA STORE (NETWORK QUERY LOG DATA) 110
Q1→ URLA, #
Q1→ URLB, #
Q2→ URLC, #
Qn→ URLM, #
INPUT MODULE 104
SYNONYM-GENERATING MODULE (SGM) 102
EXPANSION MODULE 106
DATA STORE (INITIAL CANDIDATES) 112
CLEAN-UP MODULE 114
DATA STORE (FILTERED CANDIDATES) 116
REDUCTION MODULE 118
DATA STORE (SELECTED SYNONYMS) 120
OUTPUT MODULE 122
SYNONYM-EXPANDED OUTPUT DATA
(E.G., SEE FIG. 3)
302

FIG. 1

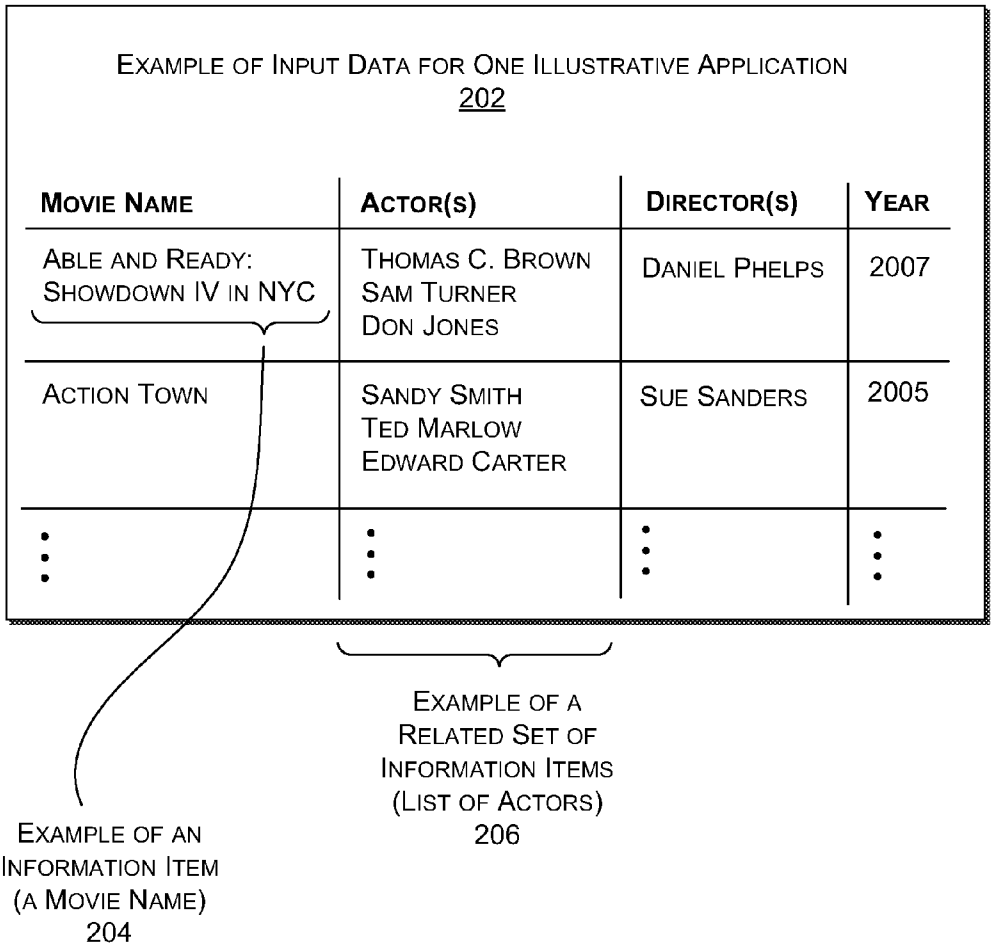

EXAMPLE OF INPUT DATA FOR ONE ILLUSTRATIVE APPLICATION
202
MOVIE NAME
ACTOR(S)
DIRECTOR(S)
YEAR
ABLE AND READY: SHOWDOWN IV IN NYC
THOMAS C. BROWN
SAM TURNER
DON JONES
DANIEL PHELPS
2007
ACTION TOWN
SANDY SMITH
TED MARLOW
EDWARD CARTER
SUE SANDERS
2005
EXAMPLE OF A RELATED SET OF INFORMATION ITEMS (LIST OF ACTORS) 206
EXAMPLE OF AN INFORMATION ITEM (A MOVIE NAME) 204

FIG. 2

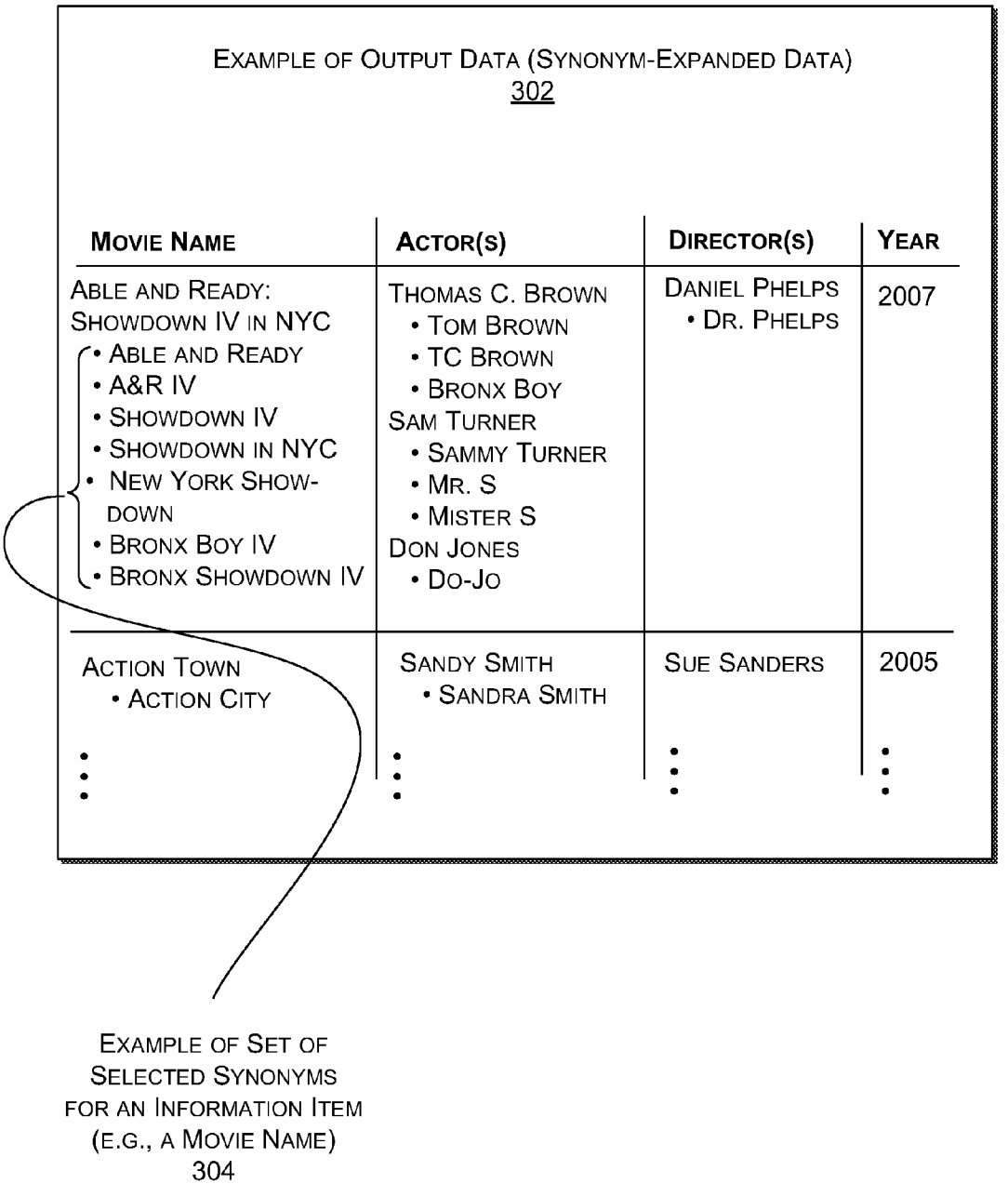

EXAMPLE OF OUTPUT DATA (SYNONYM-EXPANDED DATA)
302
MOVIE NAME
ACTOR(S)
DIRECTOR(S)
YEAR
ABLE AND READY: SHOWDOWN IV IN NYC
• ABLE AND READY
• A&R IV
• SHOWDOWN IV
• SHOWDOWN IN NYC
• NEW YORK SHOWDOWN
• BRONX BOY IV
• BRONX SHOWDOWN IV
THOMAS C. BROWN
• TOM BROWN
• TC BROWN
• BRONX BOY
SAM TURNER
• SAMMY TURNER
• MR. S
• MISTER S
DON JONES
• DO-JO
DANIEL PHELPS
• DR. PHELPS
2007
ACTION TOWN
• ACTION CITY
SANDY SMITH
• SANDRA SMITH
SUE SANDERS
2005
EXAMPLE OF SET OF SELECTED SYNONYMS FOR AN INFORMATION ITEM (E.G., A MOVIE NAME)
304

FIG. 3

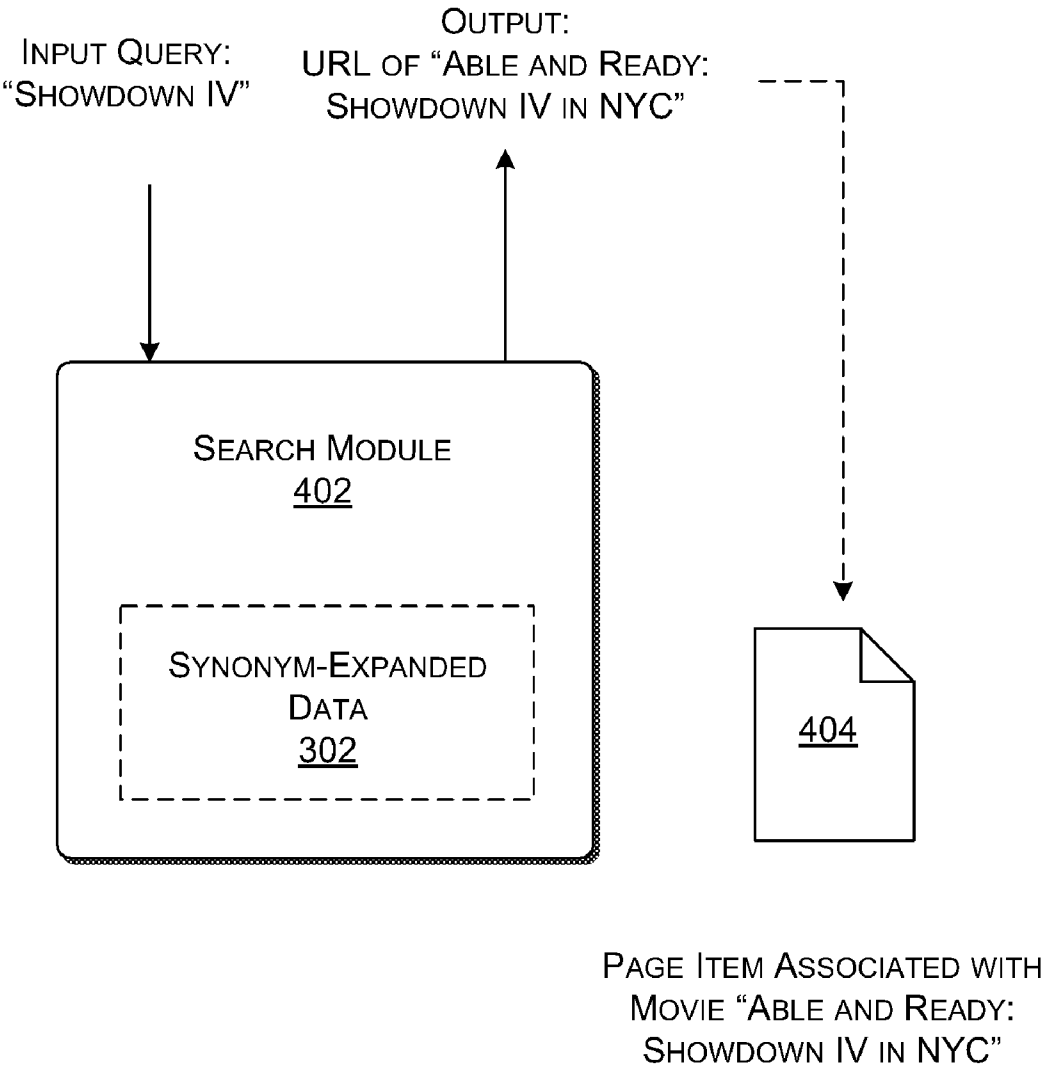

INPUT QUERY:
"SHOWDOWN IV"
OUTPUT:
URL OF "ABLE AND READY:
SHOWDOWN IV IN NYC"
SEARCH MODULE
402
SYNONYM-EXPANDED
DATA
302
404
PAGE ITEM ASSOCIATED WITH
MOVIE "ABLE AND READY:
SHOWDOWN IV IN NYC"

FIG. 4

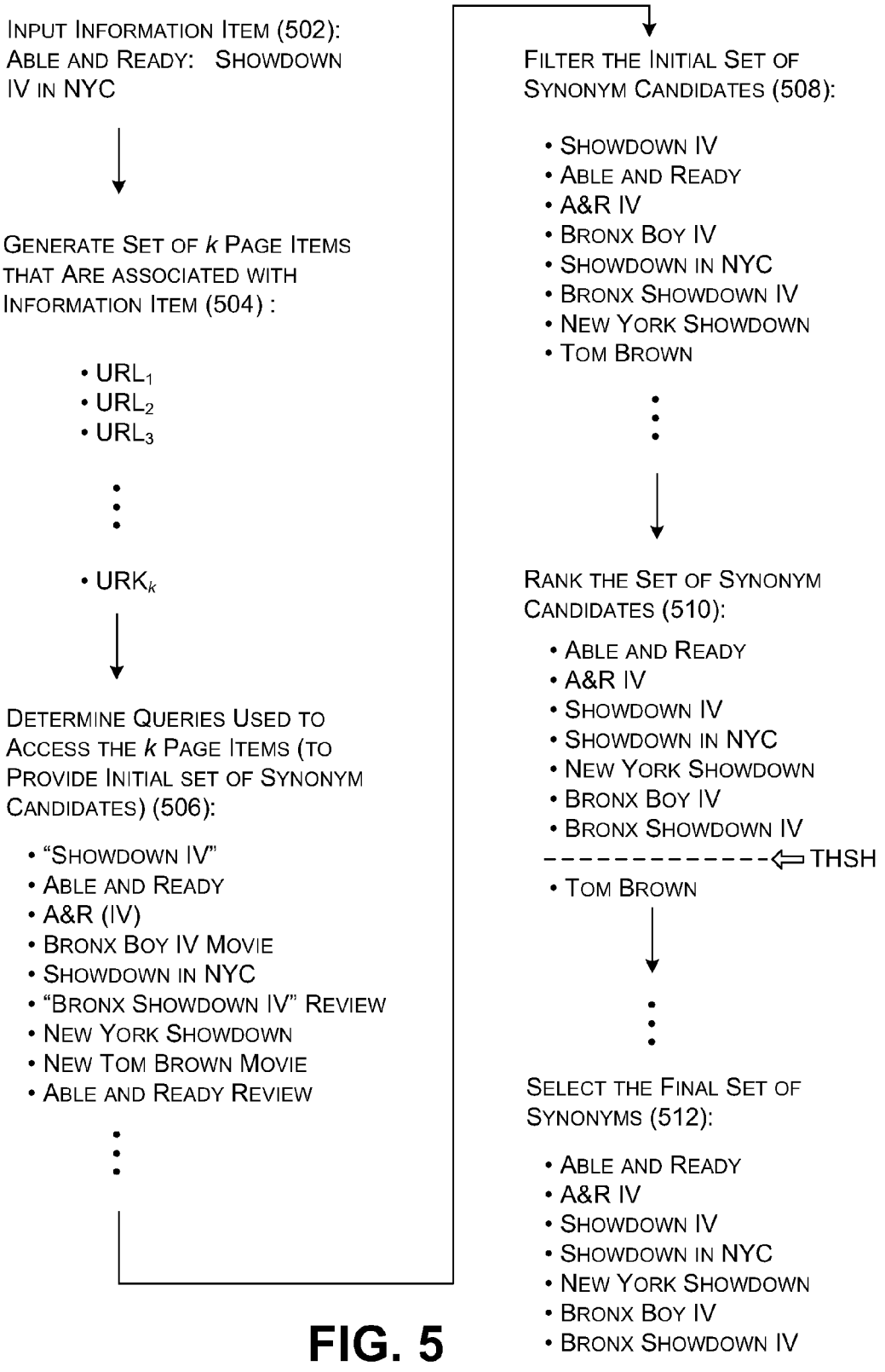

INPUT INFORMATION ITEM (502):
ABLE AND READY: SHOWDOWN IV IN NYC
GENERATE SET OF k PAGE ITEMS THAT ARE ASSOCIATED WITH INFORMATION ITEM (504) :
• URL1
• URL2
• URL3
• URLk
DETERMINE QUERIES USED TO ACCESS THE k PAGE ITEMS (TO PROVIDE INITIAL SET OF SYNONYM CANDIDATES) (506):
• "SHOWDOWN IV"
• ABLE AND READY
• A&R (IV)
• BRONX BOY IV MOVIE
• SHOWDOWN IN NYC
• "BRONX SHOWDOWN IV" REVIEW
• NEW YORK SHOWDOWN
• NEW TOM BROWN MOVIE
• ABLE AND READY REVIEW
FILTER THE INITIAL SET OF SYNONYM CANDIDATES (508):
• SHOWDOWN IV
• ABLE AND READY
• A&R IV
• BRONX BOY IV
• SHOWDOWN IN NYC
• BRONX SHOWDOWN IV
• NEW YORK SHOWDOWN
• TOM BROWN
RANK THE SET OF SYNONYM CANDIDATES (510):
• ABLE AND READY
• A&R IV
• SHOWDOWN IV
• SHOWDOWN IN NYC
• NEW YORK SHOWDOWN
• BRONX BOY IV
• BRONX SHOWDOWN IV
THSH
• TOM BROWN
SELECT THE FINAL SET OF SYNONYMS (512):
• ABLE AND READY
• A&R IV
• SHOWDOWN IV
• SHOWDOWN IN NYC
• NEW YORK SHOWDOWN
• BRONX BOY IV
• BRONX SHOWDOWN IV

FIG. 5

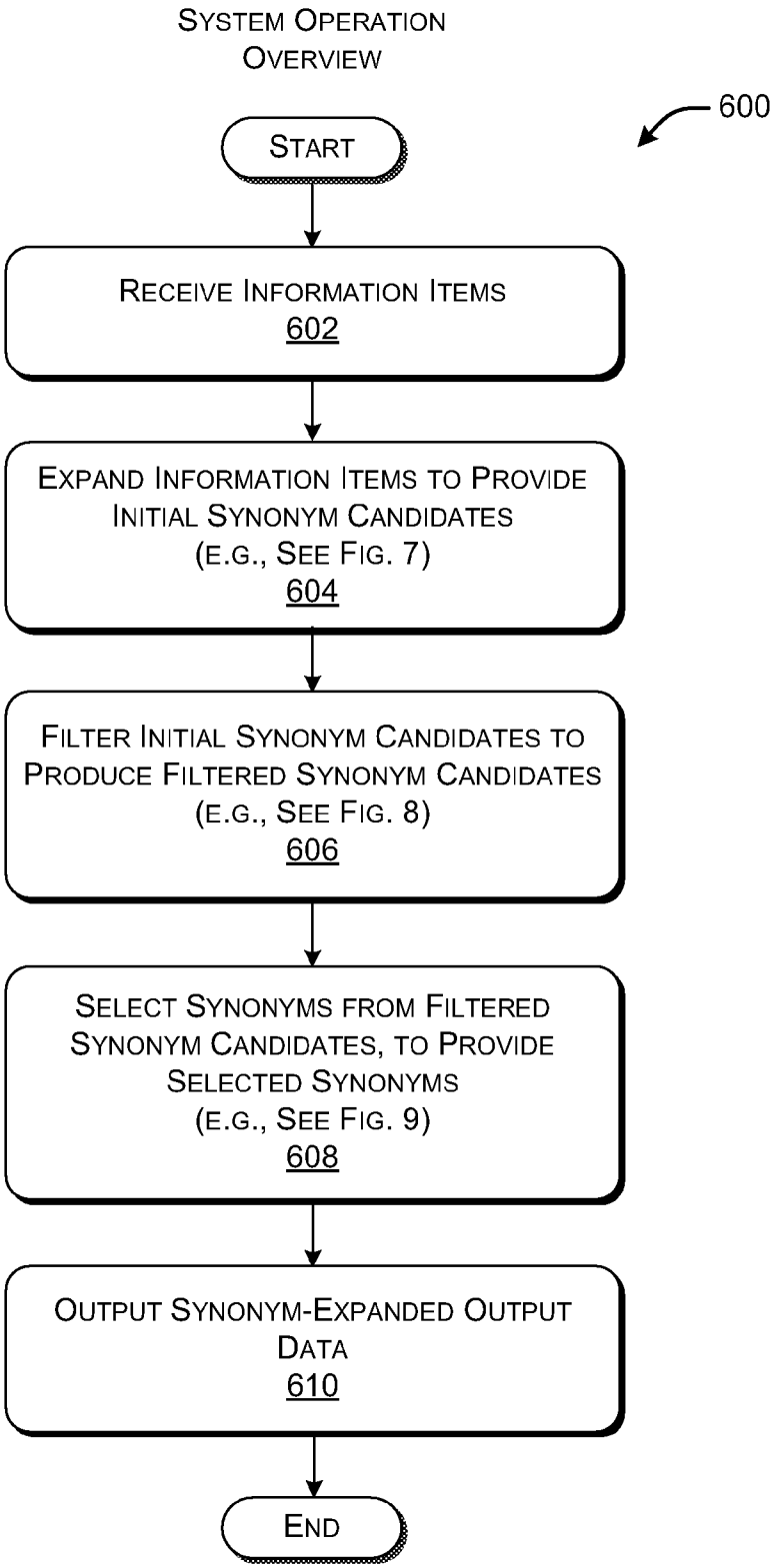

SYSTEM OPERATION OVERVIEW
600
START
RECEIVE INFORMATION ITEMS
602
EXPAND INFORMATION ITEMS TO PROVIDE INITIAL SYNONYM CANDIDATES
(E.G., SEE FIG. 7)
604
FILTER INITIAL SYNONYM CANDIDATES TO PRODUCE FILTERED SYNONYM CANDIDATES
(E.G., SEE FIG. 8)
606
SELECT SYNONYMS FROM FILTERED SYNONYM CANDIDATES, TO PROVIDE SELECTED SYNONYMS
(E.G., SEE FIG. 9)
608
OUTPUT SYNONYM-EXPANDED OUTPUT DATA
610
END

FIG. 6

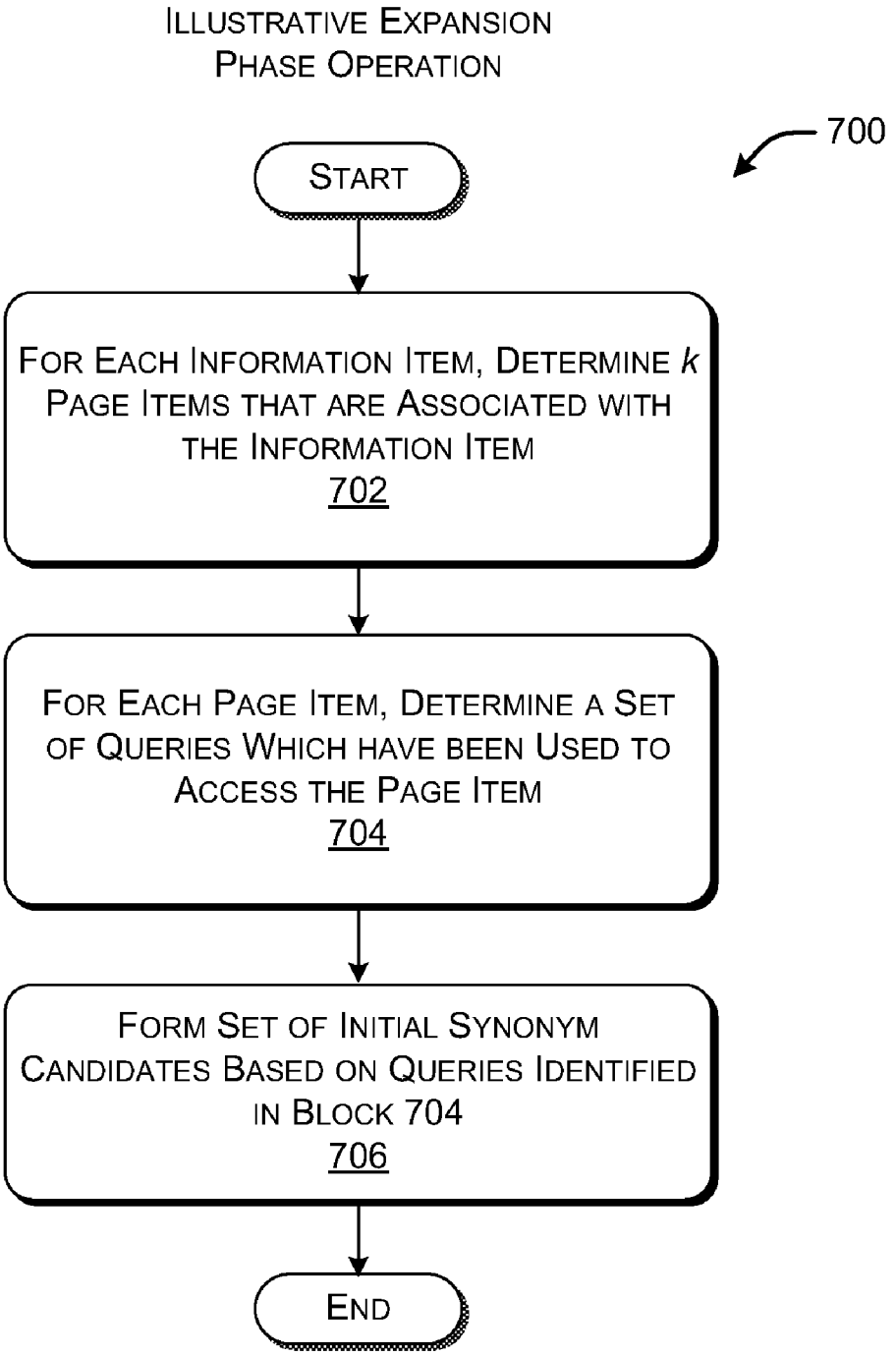

ILLUSTRATIVE EXPANSION PHASE OPERATION
700
START
FOR EACH INFORMATION ITEM, DETERMINE k PAGE ITEMS THAT ARE ASSOCIATED WITH THE INFORMATION ITEM
702
FOR EACH PAGE ITEM, DETERMINE A SET OF QUERIES WHICH HAVE BEEN USED TO ACCESS THE PAGE ITEM
704
FORM SET OF INITIAL SYNONYM CANDIDATES BASED ON QUERIES IDENTIFIED IN BLOCK 704
706
END

FIG. 7

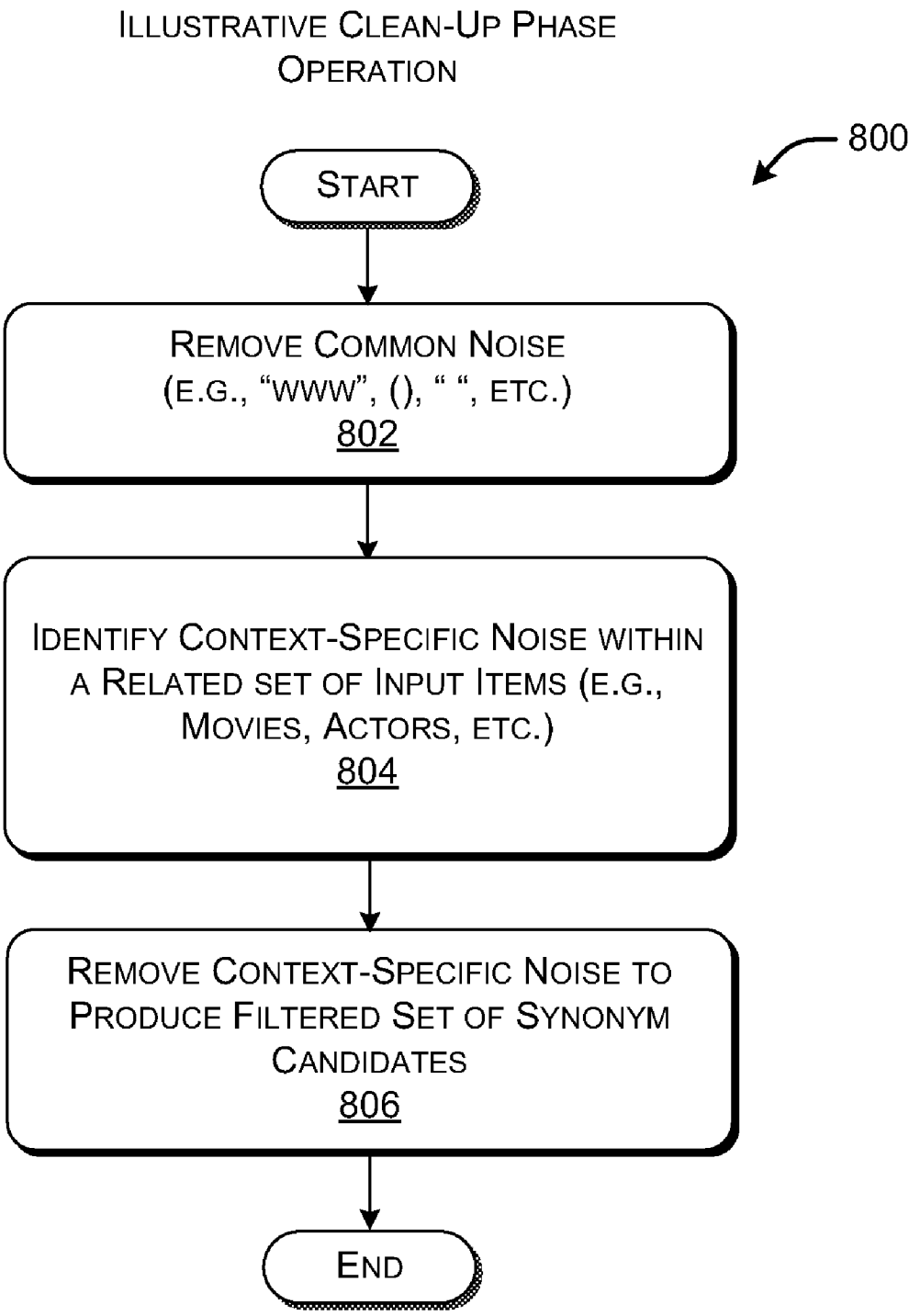

ILLUSTRATIVE CLEAN-UP PHASE OPERATION
800
START
REMOVE COMMON NOISE (E.G., "WWW", (), " ", ETC.)
802
IDENTIFY CONTEXT-SPECIFIC NOISE WITHIN A RELATED SET OF INPUT ITEMS (E.G., MOVIES, ACTORS, ETC.)
804
REMOVE CONTEXT-SPECIFIC NOISE TO PRODUCE FILTERED SET OF SYNONYM CANDIDATES
806
END

FIG. 8

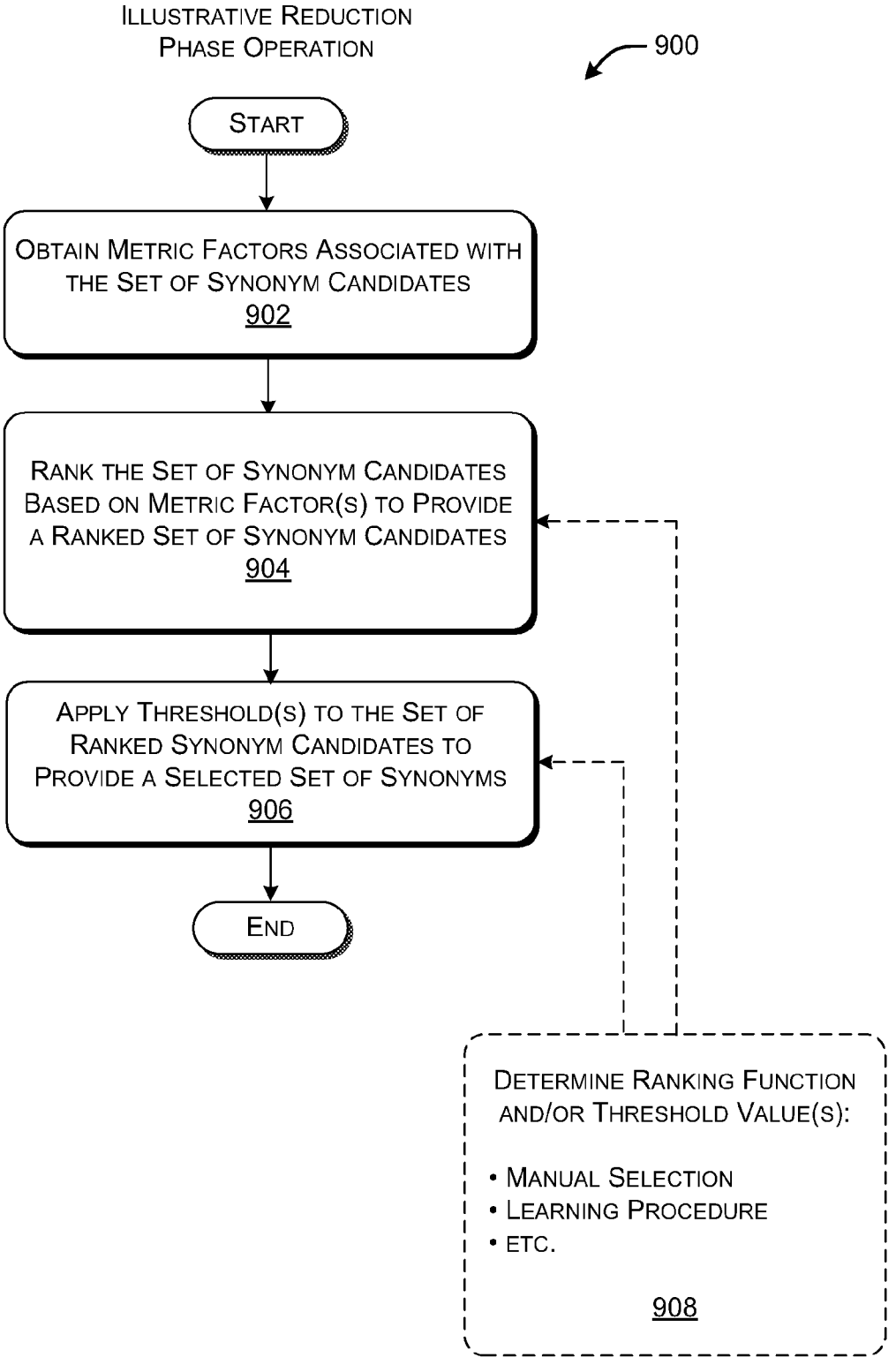

ILLUSTRATIVE REDUCTION PHASE OPERATION
900
START
OBTAIN METRIC FACTORS ASSOCIATED WITH THE SET OF SYNONYM CANDIDATES
902
RANK THE SET OF SYNONYM CANDIDATES BASED ON METRIC FACTOR(S) TO PROVIDE A RANKED SET OF SYNONYM CANDIDATES
904
APPLY THRESHOLD(S) TO THE SET OF RANKED SYNONYM CANDIDATES TO PROVIDE A SELECTED SET OF SYNONYMS
906
END
DETERMINE RANKING FUNCTION AND/OR THRESHOLD VALUE(S):
• MANUAL SELECTION
• LEARNING PROCEDURE
• ETC.
908

FIG. 9

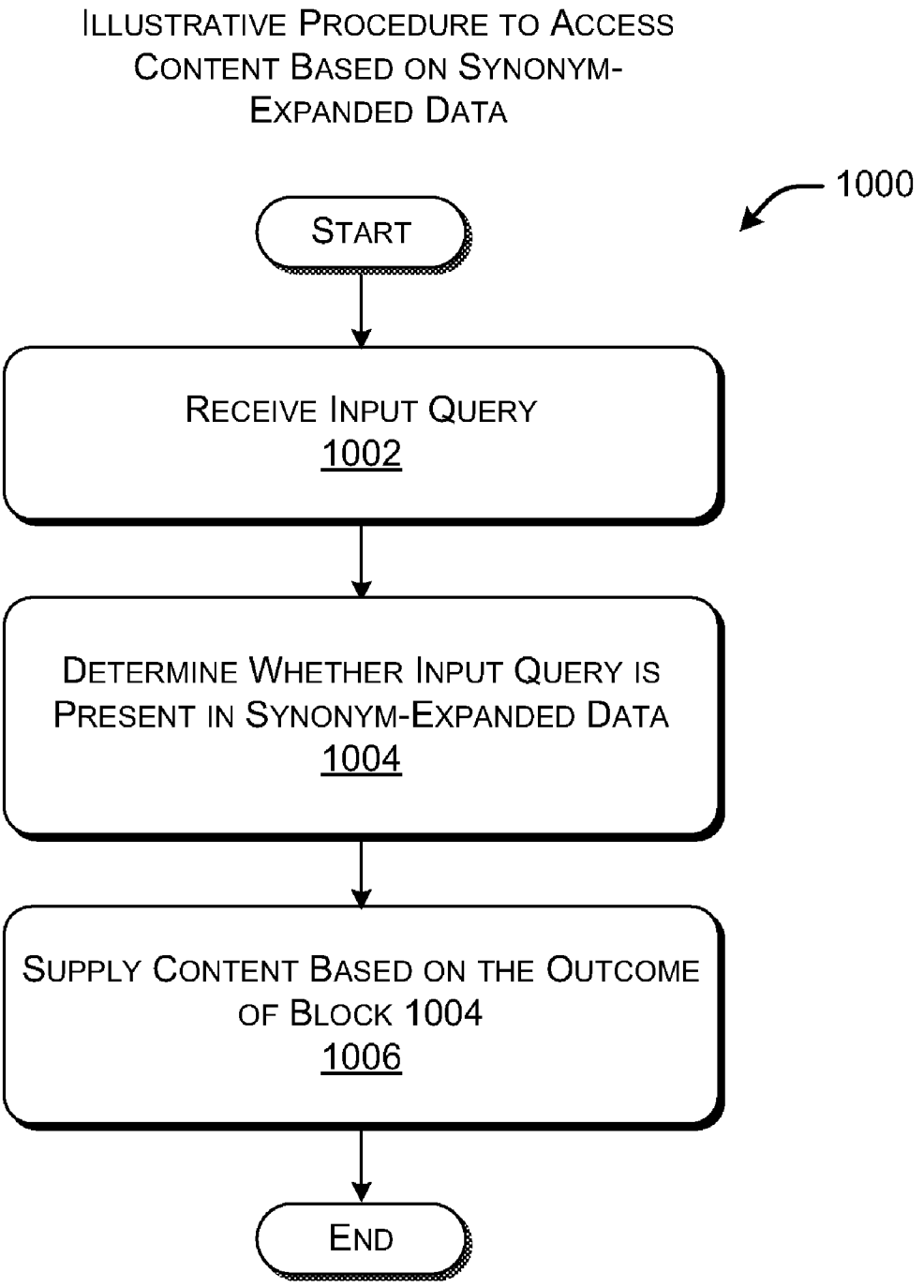

ILLUSTRATIVE PROCEDURE TO ACCESS CONTENT BASED ON SYNONYM-EXPANDED DATA
1000
START
RECEIVE INPUT QUERY
1002
DETERMINE WHETHER INPUT QUERY IS PRESENT IN SYNONYM-EXPANDED DATA
1004
SUPPLY CONTENT BASED ON THE OUTCOME OF BLOCK 1004
1006
END

FIG. 10

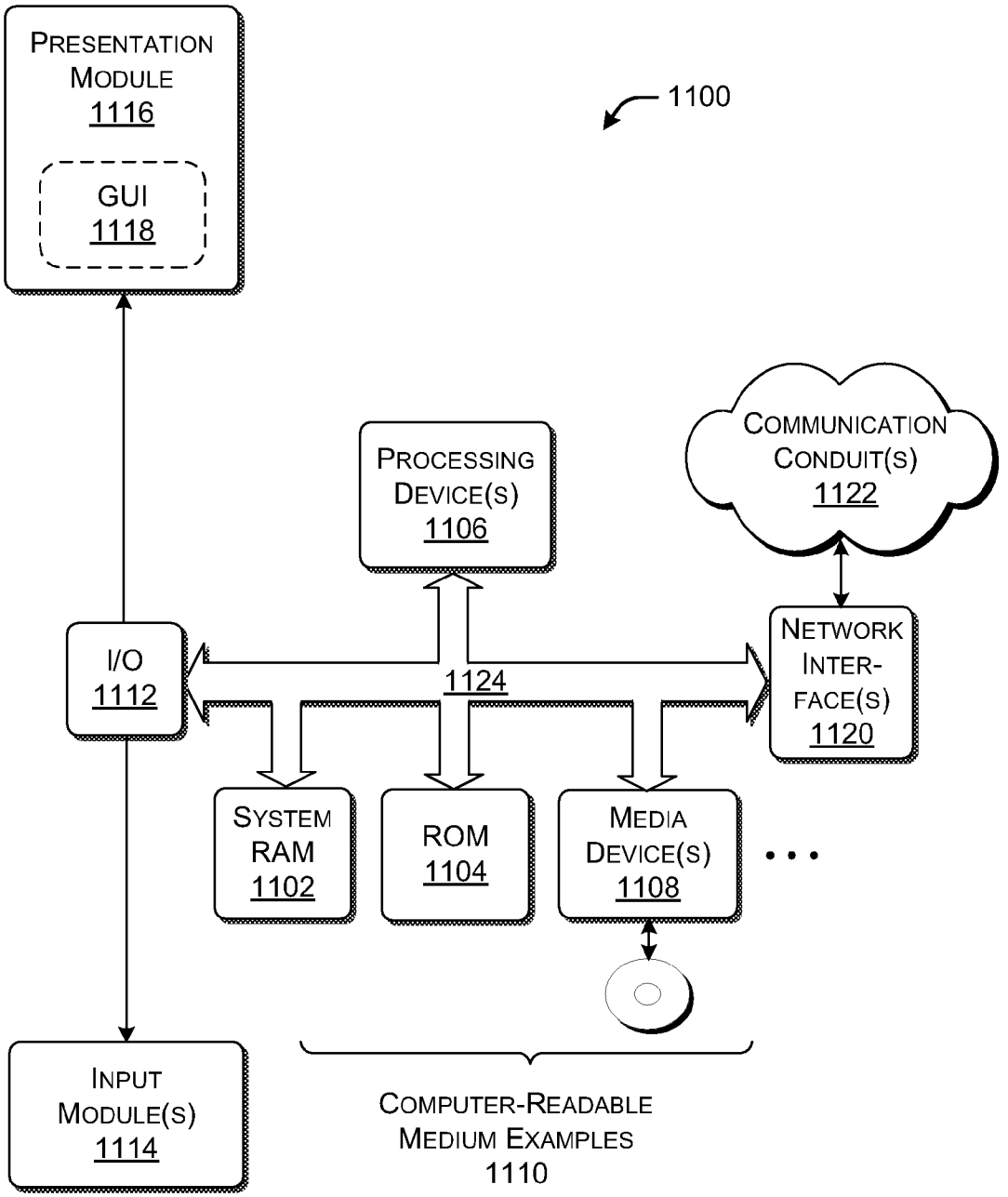

PRESENTATION MODULE 1116
GUI 1118
1100
COMMUNICATION CONDUIT(S) 1122
PROCESSING DEVICE(S) 1106
I/O 1112
1124
NETWORK INTER-FACE(S) 1120
SYSTEM RAM 1102
ROM 1104
MEDIA DEVICE(S) 1108
INPUT MODULE(S) 1114
COMPUTER-READABLE MEDIUM EXAMPLES 1110

FIG. 11

GENERATING SYNONYMS BASED ON QUERY LOG DATA

BACKGROUND

[0001] A provider of network-accessible content may provide a table which identifies salient information items pertaining to the content. For example, consider the merely illustrative case of a provider which provides content regarding movies. This provider may provide a table which identifies the movies by listing their titles, actors, directors, and so on. The information items maintained by the table may be structured (or at least partially structured) in the sense that the table can organize the information items using a defined format.

[0002] In the above example, a user who wishes to access content regarding a particular movie may submit a query which attempts to identify one or more of the information items discussed above. For example, a user may submit a query that attempts to identify the name of a desired movie, or an actor which appears in the desired movie, or a combination thereof, and so on. However, this type of retrieval tactic is not always successful. The table may identify the titles and actors of the movies using a canonical (standard) form of these entries. A user may not know the precise from in which the table stores the information items. Hence, the user may enter a query which fails to match the way information is expressed in the table. For example, the user may enter an abbreviated form of a movie title, or a nickname associated with a movie actor. This may result in the inability of the user to obtain the information that he or she is seeking.

[0003] There are known strategies for broadening a user's input query in an attempt to mitigate the above problems. For example, one known technique can identify queries which are textual variants of the query input by a user. For example, this technique may broaden an input query by removing suffixes and the like, or, more generally, by determining whether there is a matching information item that has a sufficiently small edit distance with respect to the input query. However, this type of technique may not be reliable in the above-described scenario because the common variants of the information items may have weak textual similarity (or virtually no textual similarity at all) with respect to the canonical forms of the information items. For example, the nickname of an actor may have very little textual similarity with his or her formal name. Further, the common variants can vary from the canonical forms by adding extra words, omitting words, and so on. In short, the queries entered by users may be non-trivial variations of the canonical form of the information items.

[0004] Another known technique allows a user to manually annotate a canonical form of an information item such that it includes one or more known variants. For example, a user who wishes to advertise a particular merchandise item for sale may list a set of keywords which identify the various ways that people refer to that merchandise item. However, this technique is not fully satisfactory because it requires a user to manually create and maintain the lists of variants. Further, the list of variants may fail to capture the myriad of ways in which the public refers to information items. Moreover, it is difficult to manually capture the most appropriate variants of information items because the most appropriate variants can dynamically change over time.

[0005] Known techniques for expanding information items may have yet additional shortcomings.

SUMMARY

[0006] An illustrative approach is described for generating synonyms to supplement at least one canonical information item. Users can access network-accessible content using the canonical forms of the information items or the synonyms associated with the canonical forms.

[0007] According to one illustrative aspect, the approach uses query log data to identify the synonyms. The query log data, in turn, empirically reflects the way that actual users prefer to refer to information items when making network searches. As such, the approach can improve the ability of users to access desired content, that is, by more reliably tracking the ways in which the users prefer to refer to information items. The approach can also reduce or eliminate the need for users to manually annotate information items with appropriate synonyms.

[0008] According to another illustrative aspect, the approach can operate on a set of related information items. For example, the information items may correspond to a set of items pertaining to movies, a set of items pertaining to a particular type of merchandise, and so on.

[0009] According to another illustrative aspect, the information items may be organized in the form of structured data or partially structured data.

[0010] According to another illustrative aspect, the approach can involve an expansion phase, a clean-up phase, and a reduction phase. In the expansion phase, the approach identifies, for each information item to be expanded, a set of initial synonym candidates. In the clean-up phase, the approach identifies and removes noise from the set of initial synonym candidates (if such noise is present and can be identified), to provide a set of filtered synonym candidate items. In the reduction phase, the approach ranks and applies a threshold to the set of filtered synonym candidate items, to generate, for each information item, a set of selected synonyms.

[0011] According to another illustrative aspect, the expansion phase can operate by identifying a list of k page items (e.g., Web pages) associated with each information item. For each page item, the approach can then identify a set of queries that users have used in the past to access the page item. The approach merges the queries associated with the set of k page items to provide the set of initial synonym candidates for a particular information item.

[0012] According to another illustrative aspect, the clean-up phase can operate by identifying words (corresponding to any textual features) that appear in the initial set of synonym candidates, but which do not contribute the discriminative characteristics of the synonym candidates. This information is broadly referred to herein as noise. For example, in one application, some of the synonym candidates may include the word "movie," "actor," "new," "review," and so on. These words act as context-sensitive noise that does not contribute to the uniqueness of the synonym candidates. The clean-up phase operates by identifying and removing this type of noise. In one implementation, the clean-up phase can identify the context-sensitive noise by examining a group of synonyms associated with a set of related informational items (such as a set of movie titles). The clean-up phase can operate by identifying the presence of frequently-occurring words in the

group of synonyms that do not also appear (or do not frequently appear) in the canonical forms of the information items.

[0013] According to another illustrative aspect, the reduction phase can identify various metric factors associated with each of the synonym candidates. The metric factors can be obtained from information extracted from the query log data. For each information item, the reduction phase can then use these metric factors to first rank the synonym candidates, and then apply a threshold to the synonym candidates to cull a final set of synonyms.

[0014] According to another illustrative aspect, the approach can act on a single information item, rather than a set of related information items.

[0015] This Summary is provided to introduce a selection of concepts in a simplified form; these concepts are further described below in the Detailed Description. This Summary is not intended to identify key features or essential features of the claimed subject matter, nor is it intended to be used to limit the scope of the claimed subject matter.

BRIEF DESCRIPTION OF THE DRAWINGS

[0016] FIG. **1** shows an illustrative synonym-generating module (SGM) for generating synonyms.

[0017] FIG. **2** shows an example of input data that can be operated on by the SGM of FIG. **1**.

[0018] FIG. **3** shows an example of output data that can be generated by the SGM of FIG. **1**, based on the input data shown in FIG. **2**.

[0019] FIG. **4** shows an example of how the output shown in FIG. **3** can be used to facilitate a user's attempt to access network-accessible content.

[0020] FIG. **5** shows an example of one manner in which the SGM of FIG. **1** can generate synonyms using a three-phase approach.

[0021] FIG. **6** is an illustrative procedure that describes an overview of one manner of operation of the system of FIG. **1**.

[0022] FIG. **7** is an illustrative procedure that describes one manner in which an expansion phase of the procedure of FIG. **6** can be performed.

[0023] FIG. **8** is an illustrative procedure that describes one manner in which a clean-up phase of the procedure of FIG. **6** can be performed.

[0024] FIG. **9** is an illustrative procedure that describes one manner in which a reduction phase of the procedure of FIG. **6** can be performed.

[0025] FIG. **10** is an illustrative procedure that describes how an output of the procedure of FIG. **6** can be used to facilitate a user's attempt to access network-accessible content.

[0026] FIG. **11** shows illustrative processing functionality that can be used to implement any aspect of the features shown in the foregoing drawings.

[0027] The same numbers are used throughout the disclosure and figures to reference like components and features. Series 100 numbers refer to features originally found in FIG. **1**, series 200 numbers refer to features originally found in FIG. **2**, series 300 numbers refer to features originally found in FIG. **3**, and so on.

DETAILED DESCRIPTION

[0028] An illustrative approach is described for generating synonyms to supplement at least one information item, such as, but not limited to, a set of related information items. Users can access network-accessible content using canonical forms of the information items or the synonyms associated with the information items. The approach can use a three-phase technique to identify the synonyms. The three-phase technique makes use of a query log data to identify the synonyms.

[0029] By virtue of the use of query log data, the synonyms have an increased chance of reflecting the way in which actual users prefer to refer to the information items. This, in turn, may increase the likelihood that queries submitted by the users will be successful in accessing desired network-accessible content. This advantage may apply even in those cases in which the synonyms that the users prefer to use are not textually similar to the canonical forms of the information items. The approach may further reduce or eliminate the need for users to manually create and maintain synonyms. More generally, the concepts disclosed herein may address one or more of the challenges or problems previously noted, but are not limited to addressing all or any of these challenges or problems.

[0030] This disclosure is organized as follows. Section A describes an illustrative system for generating synonyms. The synonyms can be used to facilitate retrieval of network-accessible content. Section B describes illustrative methods that explain the operation of the system of Section A. Section C describes illustrative processing functionality that can be used to implement any aspect of the features described in Sections A and B.

[0031] As a preliminary matter, some of the figures describe the concepts in the context of one or more components, variously referred to as functionality, modules, features, elements, etc. The various components shown in the figures can be implemented in any manner, for example, by software, hardware, firmware, manual processing operations, and so on, or any combination of these implementations. In one case, the illustrated separation of various components in the figures into distinct units may reflect the use of corresponding distinct physical components. Alternatively, or in addition, any single component illustrated in the figures may be implemented by plural physical components. Alternatively, or in addition, the depiction of any two or more separate components in the figures may reflect different functions performed by a single physical component. FIG. **11**, to be discussed in turn, provides additional details regarding one illustrative implementation of the functions shown in the figures.

[0032] Other figures describe the concepts in flowchart form. In this form, certain operations are described as constituting distinct blocks performed in a certain order. Such implementations are illustrative and non-limiting. Certain blocks described herein can be grouped together and performed in a single operation, certain blocks can be broken apart into plural component blocks, and certain blocks can be performed in an order that differs from that which is illustrated herein (including a parallel manner of performing the blocks). The blocks shown in the flowcharts can be implemented by software, firmware, hardware, manual processing, any combination of these implementations, and so on.

[0033] As to terminology, the phase "configured to" encompasses any way that any kind of functionality can be constructed to perform an identified operation. The functionality can be configured to perform an operation using, for instance, hardware, software, firmware, etc., and/or any combination thereof.

[0034] The term "logic" encompasses any functionality for performing a task. For instance, each operation illustrated in the flowcharts corresponds to logic for performing that operation. In one case, logic may correspond to computer-readable instructions. In another case, logic may correspond to discrete logic components, or a combination of discrete logic components and computer-readable instructions.

[0035] The term "set" encompass any collection of items, including zero items, one item, or more than one item.

[0036] A. Illustrative Systems

[0037] FIG. **1** shows a synonym-generating module (SGM) **102** for generating synonyms based on canonical forms of information items. Users can retrieve network-accessible content (e.g., network-accessible Web pages or the like) by entering the canonical forms of the information items or the synonyms.

[0038] Before discussing the individual features of the SGM **102**, this section will describe illustrative input data that can be fed to the SGM **102** and illustrative output data which can be generated by the SGM **102**, as well as processing that can be performed by the SGM **102** on the input data to transform it into the output data. FIG. **2** illustrates one example of the input data, while FIG. **3** illustrates one example of the output data. The examples shown in FIGS. **2** and **3** pertain to information items that can be used to access information regarding movies. However, the principles described with reference to these figures can be applied to any application and any context. More generally, the examples presented herein refer to one illustrative scenario in which the SGM **102** processes input data that includes multiple information items. But the principles described herein can also be applied to the case in which the SGM **102** processes a single information item.

[0039] Starting with FIG. **2**, this figure shows input data **202** that includes input information items associated with a group of movies. Each information item pertains to a salient piece of information that characterizes a movie. For example, a first column of information items identifies the names of respective movies. A second column of information items identifies the actors associated with the movies. A third column identifies the directors associated with the movies. A fourth column of information items identifies a year-of-release of the movies. Information item **204** is an example of an individual information item (in this case, corresponding to the name of a hypothetical movie entitled, "Able and Ready: Showdown IV in NYC." It will be appreciated that the selection of information items shown in FIG. **2** is merely representative, as is the organization of these information items. Further, to repeat, the SGM **102** can also act on a single information item.

[0040] The information items shown in FIG. **2** are canonical in the sense that these information items are expressed in the standard form. For example, the names of the movies correspond to the official names of the movies as specified by the respective studios which have produced the movies.

[0041] The information expressed in the input data **202** can be viewed as structured data in the sense that it is organized based on a particular format. In one implementation, the structured data can be represented by using an appropriate markup language, such as the extensible markup language (XML). In this format, a collection of appropriate tags can be used to demarcate elements of information in the input data **202** (e.g., <title>, <actor>, <director>, <year>, etc.). A schema can define how the information items are organized. The schema may optionally allow information items to be omitted or repeated. For example, the schema may accommodate multiple actor elements to be associated with a movie to account for the fact that movies include a variable number of actors. In another implementation, the input data **202** can be at least partially structured, insofar as only part of this input data **202** may be organized according to a defined structure. In another implementation, the input data **202** can be entirely unstructured.

[0042] In any case, in one implementation, the input data **202** includes one or more sets of related items. The items in a particular set are related in the sense that they pertain to a common theme (for example, the theme of movie titles for one set, the theme of actors for another set, etc.). For example, column **206** shows a set of information items that are related because they pertain to the category of actors. As will be described, in a clean-up phase, the SGM **102** can examine synonym candidates over an entire set of related items (such as over an entire set of actors). This allows the SGM **102** to identify context-sensitive noise in the set of related items that may be removed from the set synonym candidates (if such context-sensitive noise is present and can be identified).

[0043] In use, a user may enter an input query with the intent of retrieving information regarding a particular movie. The user may attempt to identify the movie by specifying one or more information items associated with the movie. For example, the user may attempt to retrieve information regarding the hypothetical movie "Able and Read: Showdown IV in NYC" by entering the title of this movie. If the user enters the name of the movie exactly as it appears in the input data **202**, then a search module can match the query with the appropriate information item in the input data **202** and provide the desired movie information to the user. However, with such a long title, the user may fail to enter the name of movie in the exact form that it is specified in the input data **202**. In this circumstance, the search module may fail to identify the correct movie in the input data **202**, and may thus potentially fail to provide the user with the desired movie information.

[0044] FIG. **3** shows illustrative output data **302** provided by the SGM **102**. In this example, the SGM **102** has supplemented the original canonical forms of the information items with synonyms of the information items. As such, the output data **302** can be viewed as a synonym-expanded version of the input data **202**.

[0045] For example, consider the information item **204** of FIG. **2**, which provides the canonical name of the longish movie title, "Able and Ready: Showdown IV in NYC." The SGM **102** identifies a set of synonyms **304**, providing the hypothetical movie title variants, "Able and Ready," "A&R IV," "Showdown IV," "Showdown in NYC," "New York Showdown," and "Bronx Boy IV." The last synonym may reflect a character name in this hypothetical movie (indicating that some members of the public may have come to associate this movie with its principal actor, rather than its proper movie title). In a similar fashion, the SGM **102** can provide synonyms for the information items in other columns of the output data **302**, such as actors, directors, and so on.

[0046] As a general feature, note that the synonyms may not represent simple textual variants of the canonical forms of the information items. For example, the actor "Sam Turner" may go by the nickname "Mr. S." This nickname is not a simple textual variant of the canonical form of the name. Further note that the synonyms can have additional words

than the canonical form of the information item, or fewer words, or the same number of words.

[0047] FIG. **4** shows how a search module **402** can use the synonym-expanded data output data **302** shown in FIG. **3** to respond to a user's search. For example, suppose that a user inputs an abbreviated name of a movie, such as "Showdown IV." The search module **402** can determine that this input query corresponds to a synonym of the movie title "Able and Ready: Showdown IV in NYC." As such, the search module **402** can provide the URL (or other identifying information) associated with the movie "Able and Ready: Showdown IV in NYC." Based on this URL, the user can retrieve a page item **404** (e.g., a Web page or the like) associated with the desired movie. As can be appreciated, the use of the synonym-expanded output data **302** allows a user to more reliably access desired movies, e.g., by reducing the chances that the search module **402** will not be able to interpret the user's query. Moreover, the SGM **102** can generate the synonyms without any manual intervention by the user, or with a reduced amount of manual intervention.

[0048] FIG. **5** illustrates one example of how the SGM **102** can generate a set of synonyms for the movie title "Able and Ready: Showdown IV in NYC." The same procedure can be performed for each of the other information items in the input data **202** (of FIG. **2**). The procedure can include three phases (or can be conceptualized to include three phases). The phases are: an expansion phase; a clean-up phase; and a reduction phase. Further, although not shown, the procedure of FIG. **5** can be applied to a single information item, rather than a set of information items.

[0049] In operation **502**, the SGM **102** identifies an information item to operate on—in this case, the movie title "Able and Ready: Showdown IV in NYC." For example, the SGM **102** can operate on the input data **202** on a column-by-column basis, processing the information items in each column in turn.

[0050] In operation **504**, the SGM **102** commences the expansion phase of the procedure. In the expansion phase, the SGM **102** expands the information item "Able and Ready: Showdown IV in NYC" into a set of initial synonym candidates. These synonyms are candidates in the sense that they have not yet been formally selected to appear in the synonym-expanded output data (e.g., in the output data **302**).

[0051] The SGM **102** can generate the set of synonym candidates in various ways. In one technique, the SGM **102** can first submit the information item to a search module as a search query. The search module can respond by returning a set of page items (such as Web pages). Each of these page items contains the information item (e.g., "Able and Ready: Showdown IV in NYC") as part thereof, or each of these page items is otherwise associated with the information item in some way. The SGM **102** can then select the top k page items in the search results returned by the search engine. More specifically, in one implementation, the search module can return a set of page items by identifying a set of URLs (or other page identifiers) that are associated with the page items. The SGM **102** can select the top k of those URLs. (But to facilitate explanation, operation **504** is described in generic terms as the identification of a set of k page items, rather than a set of k URLs.) The number k of page items that is selected can be adjusted to provide satisfactory performance based on various environment-specific factors.

[0052] The SGM **102** can obtain a list of page items in other ways, e.g., without asking a search module to perform a search. For example, the SGM **102** can identify the k page items based on query log data. The query log data identifies the queries that have been previously used by a group of users to access different respective page items (e.g., Web pages). The SGM **102** can use this query log data to map the query "Able and Ready: Showdown IV in NYC" to the page items which contain this information item or which are otherwise associated with this information item. Still other ways of identifying the k page items are possible.

[0053] In operation **506**, the SGM **102** determines, for each entry in the list of k page items, a set of queries which have been previously used to access the page item. For example, assume that one page item corresponds to a hypothetical online newspaper review of the movie "Able and Ready: Showdown IV in NYC." The SGM **102** can identify a list of queries that were used by the general public to access this newspaper review. The list of queries may contain the canonical form of the movie title as one member thereof. The list of queries may also include one or more variants of the canonical form, such as the query "Showdown IV." That is, at least one user may have accessed the newspaper article by inputting the query "Showdown IV" instead of the canonical form of the movie title. In one case, the search module can identify a potentially large number of queries that have been used to access the page item; the SGM **102** can retain a subset of these queries that are deemed to be the most viable synonym candidates. The type of considerations that can be used to assess the appropriateness of synonym candidates is described in Section B (with reference to the reduction phase of the processing).

[0054] As a result of the above-described processing being performed on all of the k page items, for each information item, the SGM **102** provides a master list of queries that have been used to access one or more of the page items in the list of k page items. The SGM **102** can remove any redundant queries that may appear in the master list. In some cases, a query in the master list may have been used to access only a single page item in the list of k page items. In other cases, a query in the master list may have been used to access two or more page items in the list of k page items. Further, the master list identifies how many times each query was used to access one or more of the k page items. The master list of queries is referred to as a set of initial synonym candidates herein. The remainder of the process examines these synonym candidates to determine whether they are indeed suitable variants for the canonical form of the information item.

[0055] The SGM **102** can use the above-described query log data to identify the queries in the initial set of synonym candidates. In one case, a search module (or multiple search modules) can be used to provide the query log data. As described above, the query log data provides a historical record of the queries that users entered over a span of time. For example, the users may correspond to members of the public. The query log data also correlates the queries with the page items that users "clicked on" (or otherwise activated) in response to their queries. The query log data can also provide count information that indicates how many instances of a particular query resulted in the activation of a particular page item.

[0056] In general terms, the approach described above has the effect of treating the k page items as proxies for the information items that appear in the input data **202**. The approach also has the effect of treating the queries made by a large group of users as a reliable indication of suitable syn-

onyms for the information items. Users may change the way in which they refer to particular information items over time. This means that the SGM **102** can dynamically change the list of synonym candidates that are deemed appropriate variants of the information items over time.

[0057] In operation **508**, the SGM **102** commences the filtering phase of the procedure. In the filtering phase, the SGM **102** removes noise from the initial set of synonym candidates (if such noise is determined to be present in the synonym candidates). Noise broadly encompasses any textual feature of the set of initial set of synonym candidates that does not contribute to the distinguishing characteristics of the synonym candidates. The outcome of the filtering phase performed by the SGM **102** is a filtered set of synonym candidates.

[0058] A first class of noise corresponds to known noise characters (e.g., common noise). This is content that is typically considered noise in all contexts. For example, as shown in operation **508**, the SGM **102** has removed the presence of quotation marks and parentheses from the initial set of synonym candidates. Other known noise characters can correspond to "www," question marks, and so on. Other common noise components can correspond to stop words, such as propositions, articles, etc. (where a "common noise component" refers to a particular feature of the common noise).

[0059] In operation **508**, the SGM **102** also identifies set-specific textual information that does not contribute to the uniqueness of the synonym candidates (referred to as context-sensitive noise herein). For example, as shown in operation **506**, the SGM **102** has removed the context-sensitive noise components "movie" and "review" from the initial set of synonym candidates (where a "context-sensitive noise component" refers to a particular feature of the context-sensitive noise). The SGM **102** can identify the presence of these context-sensitive noise components by first determining whether there are frequently-used words in a group of synonym candidates associated with a related set of information items (or some subset thereof). In the example of FIG. **5**, the SGM **102** determines whether there are commonly-used words in the group of synonym candidates associated with all of the movie titles identified in the first column of the input data **202**. The SGM **102** can then determine whether these context-sensitive noise components fail to also frequently appear in the canonical forms of the information items. For example, assume that the user inputs the query "Able and Ready Review." It is unlikely that many (if any) of the canonical forms of movie titles in the input data **202** include the word "Review." This suggests that the word "Review" has little (or no) descriptive value in identifying the desired movie. Once the SGM **102** identifies such context-sensitive noise word components, it can remove them from the list of initial set of candidate items. Note that, in this illustrative implementation, the filtering operation takes into account a list of related items, and therefore is based on a more global analysis of the information items in the input data **202** compared to the expansion phase discussed above.

[0060] By virtue of removing noise in the clean-up phase (if present), the SGM **102** creates new synonym candidates. For example, the SGM **102** can remove the word "Review" (a context-sensitive noise component) and quotations (a common noise component) from the original synonym candidate ""Bronx Showdown IV" Review" to create the new synonym candidate "Bronx Showdown IV", which does not appear in the original candidate set. Moreover, by removing the word "Review" from the original synonym candidate "Able and Ready Review", this synonym candidate now is identical to another original synonym candidate, namely "Able and Ready." The clean-up phase can address this issue by consolidating the two synonym candidates into one. Specifically, the clean-up phase can merge the statistics associated with the two original synonym candidates to provide aggregated statistical information (since the synonym candidates actually pertain to the same synonym candidate). This operation has the effect of bolstering the importance of the resultant consolidated synonym candidate ("Able and Ready") in the reduction phase (as described below).

[0061] In other cases, the SGM **102** can dispense with the filtering phase corresponding to operation **508**. If the filtering phase is not performed, the reduction phase can act directly on the output of the expansion phase. In this case, the filtered set of synonym candidates is the same as the initial set of synonym candidates (meaning that no filtering has been performed).

[0062] Alternatively, the SGM **102** can remove the common type of known noise components, without also removing the context-sensitive noise. For example, consider the case in which the SGM **102** acts on a single information item (that does not belong to a related set of information items). In this case, the SGM **102** can remove common noise from the synonym candidates, but, in one implementation, cannot remove context-sensitive noise (because there is no set of related items from which to assess the presence of context-sensitive noise). Alternatively, even in this scenario, the SGM **102** can perform some kind of global analysis by detecting that the singular information item pertains to a class of information items, and then performing the filtering analysis with respect to a representative sample of information items in this class. For example, assume that filtering analysis is to be performed on input data that consists of only one movie title. The SGM **102** can perform a lookup operation to determine that the information item is a movie title, and can then perform the filtering phase with respect to a stock group of movie titles. In other words, the other members of a related set do not need to be supplied along with the input data. In other cases, the SGM **102** can pre-determine context-sensitive noise components for different categories.

[0063] In operation **510**, the SGM **102** commences the reduction phase of the procedure. In general, one purpose of this phase is to reduce the number of synonym candidates generated in the expansion phase. The SGM **102** reduces the set of synonym candidates to a final set of synonyms that are deemed to be appropriate proxies for the canonical forms of the information items, omitting the synonym candidates that are determined to be of lesser appropriateness.

[0064] More specifically, in operation **510**, for each information item, the SGM **102** first ranks the synonym candidates in the filtered list of synonym candidates identified in the filtering phase. To this end, the SGM **102** can obtain one or more metric factors from the query log data described above. The SGM **102** can rank the synonym items on the basis of one or more of these metric factors. Section B will provide additional details regarding one way in which the SGM **102** can rank the synonym candidates.

[0065] In operation **512**, for each information item, the SGM **102** can select a predetermined number of the top-ranked synonym candidates for inclusion in a final set of synonyms. This operation can be performed by defining a threshold (THSH) for the set of synonym candidates; the

candidates that are ranked above this threshold are included, while candidates below the threshold are excluded. Again, the SGM **102** can perform this selection operation based on the metric factors obtained from the query log data. Section B will provide additional details regarding one way in which the SGM **102** can apply thresholds to the synonym candidates so as to cull a desired set of final synonyms.

[0066] With the above introduction in the form of a concrete example, the discussion now returns to the component diagram of FIG. **1**. The SGM **102** illustrated in this figure can be used to implement the example described above, but it can also be applied in many other environments, and to address many other scenarios. For instance, the SGM **102** is not limited to movie data, nor is it limited to the particular organization of data shown in FIGS. **2** and **3**. More generally, the SGM **102** can be applied to structured data, partially structured data, or entirely unstructured data. Further, the SGM **102** can be applied to processing a single information item.

[0067] The SGM **102** includes an input module **104** for receiving the input data (e.g., the data to be processed), such as input data **202** of FIG. **2**. The input module **104** can also parse the input data into sections (e.g., columns in one case), and coordinate the processing of the sections on a section-by-section basis. Alternatively, or in addition, the input module **104** can coordinate the processing on these sections so that they are processed in a parallel manner. For input that includes multiple information items, a section may contain a list of related information items, such as a list of movie titles, a list of actors, and so on.

[0068] An expansion module **106** performs the expansion phase of the synonym-related processing. That is, the expansion module **106** can expand each information item identified by the input module **104** into a set of initial synonym candidates. As explained above, the expansion module **106** can perform this task by submitting each information item (e.g., the name of a movie) to a search module **108** as a query. In response, the search module **108** can identify k page items (e.g., Web pages) that contain the information item or are otherwise associated with the information item. Then, for each page item in the list of k page items, the expansion module **106** can determine the queries that have been made (by a plurality of users) which have resulted in the activation of the page item. The expansion module **106** can perform this task by accessing a data store **110** that provides query log data. The query log data provides a history of queries submitted to the search module **108**, and also correlates the queries with the page items (e.g., Web pages) that were clicked on in response to the queries. As a result of the processing performed on the set of k page items, the expansion module **106** forms a combined set of initial synonym candidates. The expansion module **106** stores the set of initial synonym candidates in a data store **112**.

[0069] A clean-up module **114** performs the filtering phase of the synonym-related processing. That is, the clean-up module **114** identifies and removes noise in the set of initial synonym candidates. A first class of noise refers to common noise, such as quotations marks, question marks, parentheses, the letters "www," and so on. The first class can also include known stop words (e.g., articles, prepositions, etc.). A second class of noise refers to set-specific context-sensitive noise. In the manner described above, the clean-up module **114** identifies these types of context-sensitive noise components by determining the presence of frequently-occurring words in the synonym candidates that do not also add meaningful content to the synonym candidates. The second class of noise is as set-specific because the clean-up module **114** performs this clean-up analysis with respect to a group of related information items, such as a group of items in a column of the input data. Illustrative such context-sensitive noise components in the movie title column of the input data may correspond to "movie," "review," and so on. The clean-up module **114** removes the noise and stores a resultant set of filtered synonym candidates in a data store **116**. The clean-up module **114** can also merge identical synonym candidates (and associated statistics) in the manner described above.

[0070] A reduction module **118** performs the reduction phase of the synonym-related processing. The reduction module **118** can optionally perform this processing in two stages. In a first operation, for each information item, the reduction module **118** can rank the synonym candidates (in the set of filtered synonym candidates) in order of appropriateness. This creates a set of ranked synonym candidates for each information item. In a second phase, for each information item, the reduction module **118** can select a set of final synonyms from the set of ranked synonym candidates by applying one or more thresholds to the set of ranked synonym candidates. In performing these operations, the reduction module **118** can obtain and apply one or more metric factors from the query log data in the data store **110**. The manner in which the reduction module **118** operates (in one implementation) will be described in Section B below. The reduction module **118** stores the results of its ranking and threshold-based-selection in a data store **120**. That is, the data store **120** provides a final list of synonyms for each information item.

[0071] An output module **122** uses the final set of synonyms identified by the reduction module **118** to supplement the canonical forms of the information items in the input data. This operation creates synonym-expanded output data, such as the output data **302** of FIG. **3**.

[0072] The SGM **102** can perform the above-described processing based on any triggering event. In one case, the SGM **102** can perform the processing on a periodic basis (e.g., once a month). Alternatively, or in addition, the SGM **102** can perform the processing when manually instructed to do so. In some cases, it is also possible to perform the analysis in response to the user submitting a query.

[0073] In the above examples, the SGM **102** identifies synonyms for the purpose of facilitating a user's subsequent search. The SGM **102** can also be applied to other uses. For example, consider the case in which there is a significant overlap in two sets of synonyms associated with two respective information items in the input data. In this case, the SGM **102** can assume that the two information items likely describe the same information. The SGM **102** can then decide to merge the two information items into a single entry. Generally, the SGM **102** can apply the above-type of analysis to identify redundant entries in the input data and to subsequently eliminate such entries. Still other applications of the SGM **102** are possible.

[0074] B. Illustrative Processes

[0075] FIGS. **6-10** describe the operation of the synonym-generating module (SGM) **102** of FIG. **1** in flowchart form. Since the principles underlying the operation of the SGM **102** have already been described in Section A, this section will serve mostly as a summary of the operation of the SGM **102**, except with respect to FIG. **9** which provides additional details regarding the operation of the reduction phase of the procedure.

[0076] FIG. **6** is an illustrative procedure **600** which provides an overview of the overall process performed by the SGM **102**. As stated, in one implementation, the procedure **600** includes three phases (an expansion phase, a clean-up phase, and a reduction phase).

[0077] In block **602**, the SGM **102** receives input data, which can represent structured data, partially structured data, or unstructured data. The data can include one or more related sets of information items, such as a list of movies, a list of actors, and so on. Or the input data can include a single information item.

[0078] In block **604**, the SGM **102** expands each information item in the input data into a set of initial synonym candidates. This operation corresponds to the expansion phase of the procedure **600**, which is described in greater detail in FIG. **7**.

[0079] In block **606**, the SGM **102** filters the initial synonym candidates to remove noise from these synonym candidates. This operation corresponds to the clean-up phase of the procedure **600**, and is described in greater detail in FIG. **8**.

[0080] In block **608**, the SGM **102** reduces the synonym candidates to a set of appropriate synonyms (e.g., by ranking the synonyms and applying one or more thresholds to the synonyms). This operation corresponds to the reduction phase of the procedure **600**, which is described in greater detail in FIG. **9**.

[0081] In block **610**, the SGM **102** uses the synonyms identified in the reduction phase to create the synonym-expanded output data. FIG. **3** provides an example of such synonym-expanded output data.

[0082] FIG. **7** is an illustrative procedure **700** which provides further information regarding the expansion phase of the overall process **600**.

[0083] In block **702**, the SGM **102** determines k page items (e.g., Web pages) that are associated with each information item in the input data. One way that the SGM **102** can perform this task is to submit each information item as a query to the search module **108**, upon which it receives k page items which contain this information item or are otherwise associated with the information item. More specifically, the search module **108** can return a list of k URLs (or other network addresses) which respectively identify the page items.

[0084] In block **704**, for each of the k page items, the SGM **102** determines a set of queries that are associated with the page item. In one case, these queries correspond to search queries that users have previously submitted to access the page item, as revealed by the query log data. More specifically, for each page item, the SGM **102** can retain a subset of the identified queries identified by the query log data. The SGM **102** can retain that subset of queries which are most strongly linked to the activation of the page item. The SGM **102** can make this determination, in turn, based on one or more metric factors obtained from the query log data. FIG. **9** describes processing of metric factors that can be performed in the context of the reduction phase; related analysis can be performed in the present context of the expansion phase to select a group of queries from a larger group of queries.

[0085] In block **706**, for each information item, the SGM **102** forms a set of initial synonym candidates by combining the queries identified in block **704** for all of the k page items. In other words, the collected queries correspond to synonym candidates.

[0086] FIG. **8** is an illustrative procedure **800** which provides further information regarding the clean-up phase of the overall process **600**.

[0087] In block **802**, the SGM **102** removes common noise from the set of initial synonym candidates, such as parentheses, quotation marks, question marks, the characters "www," stop words, and so on.

[0088] In block **804**, the SGM **102** identifies and removes context-sensitive noise components in the set of initial synonym candidates. The SGM **102** performs this analysis with respect to a group of related items, such as movies, actors, and so on.

[0089] In block **806**, the SGM **102** generates a set of filtered synonym candidates. This set is the same as the initial set of synonym candidates, except that the identified noise components have been removed. Although not shown, the SGM **102** can also merge identical synonym candidates (and associated statistics) in the manner described above.

[0090] FIG. **9** is a procedure **900** which provides further information regarding the reduction phase of the overall process **600**.

[0091] In block **902**, the SGM **102** obtains metric factors associated with each synonym candidate. In one case, the SGM **102** can obtain these metric factors from the query log data maintained by the search module **108** in the data store **110**. In some cases, the SGM **102** can obtain the desired metric factors directly from the query log data. In other cases, the SGM **102** can derive the desired metric factors by performing calculations on the query log data.

[0092] One type of metric factor that can be derived from the query log data corresponds to a log likelihood query factor. This metric factor can be calculated as follows. First, the probability that synonym candidate l (associated with a query l) is a synonym of a value v, observing document d in a collection D_v, can be provided by:

$$P(l \simeq v \mid d) = \frac{f_{ld}}{\sum_{q \in Q_d} f_{qd}}. \quad (1)$$

[0093] More specifically, D_v refers to a collection of documents associated with a value v. For example, D_v could correspond to the k page items obtained in block **702** associated with the value v (where the value v corresponds to the canonical form of the information item). Further, Q_d refers to all of the queries that result in a click on document d, f_{ld} refers to a number of times document d is clicked for query l, and f_{qd} refers to a number of times document d is clicked for query q. In effect, equation (1) states that the probability of query l being a synonym for v, given document d, is proportional to the number of times that query l results in a click on document d.

[0094] To avoid zero probabilities for queries that did not result in clicking on document d, smoothing can be performed to distribute the probabilities to such queries. Interpolation-based smoothing can be performed using the following equation:

$$\hat{P}(l \simeq v \mid d) = \lambda P(l \simeq v \mid d) + (1-\lambda) P(l \simeq v \mid CL) \quad (2).$$

[0095] Here, λ refers to a coefficient controlling the probability mass assigned to unclicked queries, and P(l≃v|CL)

refers to the background probability that l is clicked within an entire query log CL. The background probability can be calculated as:

$$P(l \simeq v \mid CL) = \frac{f_l}{\sum_{q \in Q_d} f_q} \quad (3)$$

[0096] Here, f_q refers to a number of times query q is asked in click log CL. f_l refers to the number of times query l is asked in click log CL.

[0097] The probability of query l being a synonym for value v over the observed document collection D_v is the joint probability of the probabilities over the documents in D_v. The joint probability can be calculated by:

$$P(l \simeq v|D_v) = \Pi_{d \in D_v} \hat{P}(l \simeq v|d) \quad (4).$$

[0098] Applying the log function over the two sides of equation (4) results in the log likelihood:

$$\log P(l \simeq v|D_v) = \Sigma_{d \in D_v} \log \hat{P}(l \simeq v|d) \quad (5).$$

[0099] This log likelihood can be used as the empirical log likelihood that l is a synonym for value v, that is:

$$\log P(l \simeq v) \approx \log P(l \simeq v|D_v) \quad (6).$$

[0100] Another type of metric factor that the SGM **102** can obtain corresponds to an indication of a total number times that a particular synonym candidate has been used to access any one of the k page items.

[0101] Another type of metric factor corresponds to an indication of how many of the k page items are associated with a particular synonym candidate.

[0102] Another type of metric factor corresponds to a total number of times that a particular synonym candidate has been use to access any page item (that is, not just the k page items).

[0103] Another type of metric factor corresponds to a total number of times that users were given an opportunity to click on one or the k page items upon submitting a particular query (regardless of whether the users actually clicked on one of the k page items). In the context of a typical search module, this type of impression-related metric factor can identify how many times one of the k page items appeared in the search results in response to entering a particular synonym candidate as a search term.

[0104] Still other types of metric factors can be obtained.

[0105] In block **904**, the SGM **102** ranks the synonym candidates based on one or more of the metric factors identified above (or some other type of metric factor or combination thereof). The SGM **102** can apply various environment-specific techniques to rank the synonym candidates. In one technique, the SGM **102** can rely on a single metric factor to rank the synonym candidates, such as log likelihood. In this case, the SGM **102** can rank the synonym candidates from most appropriate to least appropriate based on the log likelihood values associated with the synonym candidates.

[0106] In other cases, the SGM **102** can rely on a combination of two or more metric factors to rank the synonym candidates based on any type of function or consideration, including any type of linear function, any type of non-linear function, and so on. For example, in one case, the SGM **102** can assign a score to each synonym candidate that is based on a weighted linear combination of two or more metric factors.

[0107] In any of the cases described above, the SGM **102** can rely on different considerations to select and configure the ranking function applied to rank the synonym candidates. In one case, the SGM **102** can rely on a user to manually examine the metric factors associated with a list of synonym candidates and select a ranking function that is deemed suitable to rank the synonym candidates. For example, the user may examine the metric factors to determine that the log likelihood is a suitable metric factor for use in ranking the synonym candidates.

[0108] In another case, the SGM **102** can apply an automated or semi-automated procedure to identify a suitable ranking function for use in ranking the synonym candidates. For example, in a separate learning procedure, a user can examine a training set of synonyms and associated metric factors. Based on this analysis, the user can label the synonyms in this training set to reflect whether they are appropriate or inappropriate variants of the canonical forms of the associated information items (such as movie titles). Or the user can assign variable scores (e.g., from 0 to 1) to the synonyms depending their assessed levels of appropriateness. Based on the labels applied by the user, the SGM **102** can then perform a training operation which automatically identifies the metric factors which are capable of discriminating suitable synonym candidates from unsuitable synonym candidates. Based on this analysis, the SGM **102** can select the metric factor (or metric factors) to be used to rank synonym candidates. Alternatively, or in addition, if it is deemed appropriate to use a multi-metric function to rank the synonym candidates, the SGM **102** can use the above-described learning procedure to calculate the weights to be applied to the individual metric factors within the function.

[0109] In any case, the SGM **102** can apply different consideration (and different functions) to rank synonym candidates associated with different sets of related items. For example, the SGM **102** may determine that a first type of function is appropriate to rank synonym candidates associated with movie titles, while a second type of function is appropriate to rank synonym candidates associated with actors, and so on. Further, the SGM **102** may determine that the appropriateness of a ranking function may change over time.

[0110] In block **906**, the SGM **102** applies a threshold (or thresholds) to each set of ranked synonym candidates associated with an information item. The SGM **102** selects the synonym candidates above such a threshold as the final set of synonyms to be used in the synonym-expanded output data, discarding the remainder. Again, the SGM **102** can use various techniques to determine the threshold to be used. In one case, the SGM **102** can rely on a human user to examine the ranked list of synonym candidates and select an appropriate threshold. For example, the user can assign a threshold that corresponds to a point at which the metric factor values appear to markedly drop (if such a dramatic drop-off point in fact exists).

[0111] In another case, the SGM **102** can apply the type of automated or semi-automated learning procedure described above to identify appropriate thresholds. For example, the user can manually label a training set of synonyms to indicate their respective appropriateness. The SGM **102** can then apply a training procedure to automatically determine a threshold value that can be used to demarcate appropriate synonyms from inappropriate synonyms.

[0112] In general, block **908** of FIG. **9** identifies any type of manual analysis and/or automated learning analysis that can be used to define and configure the analysis tools used in block **904** and/or block **906**, as described above.

[0113] FIG. **10** is an illustrative procedure **1000** which explains how the synonym-expanded output data can be used to facilitate a user's search for network-accessible content.

[0114] In block **1002**, a search module (such as search module **402** in FIG. **4**) receives an input query from a user. The input query may or may not match a canonical form of an information item maintained by the search module **402**.

[0115] In block **1004**, the search module **402** determines whether the input query matches any information item in the synonym-expanded data. The search module **402** can identify a match if the input query matches a canonical form of an information item or a synonym counterpart of the canonical form.

[0116] In bock **1006**, if there is a match, the search module **402** can supply information regarding the matching entry to the user. The user can use this information to retrieve the desired network-accessible content (e.g., a Web page or the like).

[0117] C. Representative Processing Functionality

[0118] FIG. **11** sets forth illustrative electrical data processing functionality **1100** (simply "processing functionality" below) that can be used to implement any aspect of the functions described above. With reference to FIG. **1**, for instance, the type of processing functionality **1100** shown in FIG. **11** can be used to implement any aspect of the synonym-generating module (SGM) **102**. In one case, the processing functionality **1100** may correspond to any type of computing device.

[0119] The processing functionality **1100** can include volatile and non-volatile memory, such as RAM **1102** and ROM **1104**, as well as one or more processing devices **1106**. The processing functionality **1100** also optionally includes various media devices **1108**, such as a hard disk module, an optical disk module, and so forth. The processing functionality **1100** can perform various operations identified above when the processing device(s) **1106** executes instructions that are maintained by memory (e.g., RAM **1102**, ROM **1104**, or elsewhere). More generally, instructions and other information can be stored on any computer-readable medium **1110**, including, but not limited to, static memory storage devices, magnetic storage devices, optical storage devices, and so on. The term "computer-readable medium also encompasses plural storage devices. The term computer-readable medium also encompasses signals transmitted from a first location to a second location, e.g., via wire, cable, wireless transmission, etc.

[0120] The processing functionality **1100** also includes an input/output module **1112** for receiving various inputs from a user (via input modules **1114**), and for providing various outputs to the user (via output modules). One particular output mechanism may include a presentation module **1116** and an associated graphical user interface (GUI) **1118**. The processing functionality **1100** can also include one or more network interfaces **1120** for exchanging data with other devices via one or more communication conduits **1122**. One or more communication buses **1124** communicatively couple the above-described components together.

[0121] In closing, the description may have described various concepts in the context of illustrative challenges or problems. This manner of explication does not constitute an admission that others have appreciated and/or articulated the challenges or problems in the manner specified herein.

[0122] More generally, although the subject matter has been described in language specific to structural features and/or methodological acts, it is to be understood that the subject matter defined in the appended claims is not necessarily limited to the specific features or acts described above. Rather, the specific features and acts described above are disclosed as example forms of implementing the claims.

What is claimed is:

1. A method for generating synonyms using data processing functionality, comprising:
- receiving a set of input information items;
- for each information item in the set of input information items:
  - generating a set of initial synonym candidates based on query log data;
  - removing noise from the set of initial synonym candidates, if said noise is present and can be identified, an output of said generating and removing comprising a set of filtered synonym candidates; and
  - reducing the set of filtered synonym candidates to a set of selected synonyms; and
- outputting synonym-expanded data that is formed based on said reducing performed with respect to each information item in the set of input information items.

2. The method of claim **1**, wherein the set of input information items comprises a list of canonical information items associated with a common theme.

3. The method of claim **1**, wherein the set of input information items comprises at least part of a collection of data, the collection of data being at least partially structured.

4. The method of claim **1**, wherein the set of input information items comprises a single information item, and wherein no noise is removed from the set of initial synonym candidates, making the set of initial synonym candidates the same as the set of filtered synonym candidates.

5. The method of claim **1**, wherein said generating comprises:
- identifying, for each information item, a set of k associated page items;
- for each page item in the set of k page items, identifying, based on the query log data, a set of associated queries; and
- providing, for each information item, the set of initial synonym candidates by combining sets of queries identified for respective page items in the set of k page items.

6. The method of claim **5**, wherein said identifying a set of k page items comprises:
- submitting each information item to a search module;
- receiving, for each information item, a set of associated initial page items; and
- selecting, for each information item, the k page items from the set of associated initial page items.

7. The method of claim **1**, wherein said removing comprises:
- identifying, for each information item, at least one context-sensitive noise component within the set of initial synonym candidates, if said at least one context-sensitive noise component is present and can be identified, said at least one context-sensitive noise component affecting synonym candidates associated with two or more information items in the set of input information items; and

removing said at least one context-sensitive noise component from the set of initial synonym candidates.

8. The method of claim **1**, wherein said reducing comprises:

obtaining at least one metric factor associated with the set of filtered synonym candidates for each information item, based on the query log data;

ranking the set of filtered synonym candidates based on said at least one metric factor to provide a set of ranked synonym candidates; and

selecting the set of selected synonyms from the set of ranked synonym candidates based on said at least one metric factor.

9. The method of claim **8**, wherein said selecting of the set of selected synonyms comprises applying a threshold to the set of ranked synonym candidates to identify the set of selected synonyms.

10. The method of claim **1**, wherein the synonym-expanded data comprises the set of input information items in canonical form, supplemented to include, for each input information item, an associated set of selected synonyms.

11. A synonym-generating module, comprising:

an input module configured to receive a set of input information items;

an expansion module configured to expand each information item in the set of input information items into a set of initial synonym candidates based on query log data, the query log data reflecting associations between prior queries submitted by users and page items accessed by the users in response to the submitted queries;

a clean-up module configured to remove, for each information item, noise from the set of initial synonym candidates, if said noise is present and can be identified, an output of the expansion module and the clean-up module comprising a set of filtered synonym candidates; and

a reduction module configured to select, for each information item, a set of synonyms from the set of filtered synonym candidates based on the query log data, to provide a set of selected synonyms; and

an output module configured to output synonym-expanded data that is formed based on the selecting performed by the reduction module with respect to each information item in the set of input information items.

12. The synonym-generating module of claim **11**, wherein the set of input information items comprises a list of canonical information items associated with a common theme.

13. The synonym-generating module of claim **11**, wherein the set of input information items comprises a single information item, and wherein no noise is removed from the set of initial synonym candidates by the clean-up module, making the set of initial synonym candidates the same as the set of filtered synonym candidates.

14. The synonym-generating module of claim **11**, wherein the expansion module comprises:

logic configured to identify, for each information item, a set of k associated page items;

logic configured to identify, for each page item in the set of k page items, a set of associated queries based on the query log data; and

logic configured to provide, for each information item, the set of initial synonym candidates by combining sets of queries identified for respective page items in the set of k page items.

15. The synonym-generating module of claim **14**, wherein said logic configured to identify a set of k page items comprises:

log configured to submit each information item to a search module;

logic configured to receive, for each information item, a set of associated initial page items; and

logic configured to select, for each information item, the k page items from the set of associated initial page items.

16. The synonym-generating module of claim **11**, wherein said clean-up module comprises:

logic configured to identify, for each information item, at least one context-sensitive noise component within the set of initial synonym candidates, if said at least one context-sensitive noise component is present and can be identified, said at least one context-sensitive noise component affecting synonym candidates associated with two or more information items in the set of input information items; and

logic configured to remove said at least one context-sensitive noise component from the set of initial synonym candidates.

17. The synonym-generating module of claim **11**, wherein said reduction module comprises:

logic configured to obtain at least one metric factor associated with the set of filtered synonym candidates for each information item, based on the query log data;

logic configured to rank the set of filtered synonym candidates based on said at least one metric factor to provide a set of ranked synonym candidates; and

logic configured to select the set of selected synonyms from the set of ranked synonym candidates based on said at least one metric factor.

18. The synonym-generating module of claim **17**, wherein said logic configured to select the set of selected synonyms comprises logic configured to apply a threshold to the set of ranked synonym candidates to identify the set of selected synonyms.

19. The synonym-generating module of claim **11**, wherein the synonym-expanded data comprises the set of input information items in canonical form, supplemented to include, for each input information item, an associated set of selected synonyms.

20. A computer-readable medium for storing computer-readable instructions, the computer-readable instructions providing a synonym-generating module when executed by one or more processing devices, the computer-readable instructions comprising:

logic configured to receive at least partially structured data, said at least partially structured data including a set of input information items, the set of input information items including one or more information items; and

logic configured to generate, for each information item in the set of input information items, a set of selected synonyms based on query log data, the query log data reflecting associations between prior queries submitted by users and page items accessed by the users in response to the submitted queries.

* * * * *